

EP0961197

Publication Title:

A user interface mechanism for manipulating context in computer management applications

Abstract:

Contextual information that is presented to the user of a windows-based computer environment through the user interface of a single open window of the environment may be readily filtered and changed by appropriate manipulation by the user of a context control feature (12) of the user interface. Manipulation of the context control feature (12) allows the user to look at a given set of objects or tasks from different views or perspectives. It also provides the user with access to a different set of objects, thereby allowing the user to look at a different set of objects or tasks from the same view or perspective as an earlier set of objects or tasks.

Data supplied from the esp@cenet database - <http://ep.espacenet.com>

EUROPEAN PATENT APPLICATION

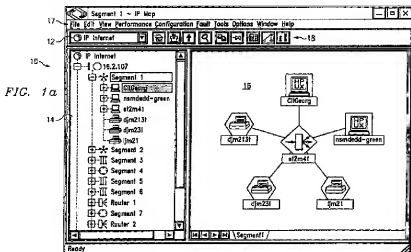
- (51) Int. Cl.⁶: G06F 3/033

- Besaw, Lawrence M.
 - Fort Collins, CO 80526 (US)
 - Raymond, Robert
 - Fort Collins, CO 80526 (US)
 - Anschpach, Mark S.
 - Fort Collins, CO 80526 (US)
 - Webb, Jayson M.
 - Boulder, CO 80302 (US)
- 1) Representative:
- Schoppe, Fritz, Dipl.-Ing.
- Schoppe, Zimmermann & Stöckeler
- Patentanwälte
- Postfach 71 08 67
- 81458 München (DE)

(54) **A user interface mechanism for manipulating context in computer management applications**

(57) Contextual information that is presented to the user of a windows-based computer environment through the user interface of a single open window of the environment may be readily filtered and changed by appropriate manipulation by the user of a context control feature (12) of the user interface. Manipulation of the context control feature (12) allows the user to look at a

given set of objects or tasks from different views or perspectives. It also provides the user with access to a different set of objects, thereby allowing the user to look at a different set of objects or tasks from the same view or perspective as an earlier set of objects or tasks.



Description

FIELD OF THE INVENTION

[0001] The present invention relates generally to windows-based computer applications, and more particularly to a user interface mechanism for manipulating context in windows-based computer applications, such as network and systems applications.

BACKGROUND OF THE INVENTION

[0002] Many computer environments employ windows having graphical user interface (GUI) capabilities that aid the user in interacting with various applications in a straight-forward and intuitive manner. Whether the user is operating a stand-alone computer, a networked computer, or other equipment in the computer environment, graphical user interfaces presented in windows have become the environment of choice for many users rather than command line interfaces. The windows-based computer environment enables the user to access any number of computer applications simply by opening an appropriate window for the desired application.

[0003] Typically a user operating in a windows-based computer environment must open a new window each time that a new application is to be used. Many applications do not have all of the functionality needed to accomplish all aspects of a user's job (e.g. editing graphics). Assume that a user has two graphical editing applications, one application provides editing capabilities for vector graphics and the other provides editing of pixel-based graphics. If the user has been editing vector graphics with one application and now wants to edit pixel-based graphics, the user must first open a second graphics editing application in a new window. It is not currently possible to access the pixel-based graphics editing application from the open application for editing vector graphics since a separate window must be opened for each application. There is therefore a requirement of a one-to-one correspondence between a computer application and a window that supports the application. For anyone using a sizable number of windows-based applications, this means that the user must open a great number of windows.

[0004] A disadvantage of this requirement is that any open window must be managed by a window manager of the graphical user interface (GUI) that produces and drives windows on a display. A window manager is a software program or module that defines the parameters of windows and generates the windows. The more windows that are open in a computer environment, the more that overhead associated with window manager programs will slow down the operation of the environment. The excess of windows also creates cognitive overhead for the user in that they have to deal with minimizing, restoring, and moving windows to be able to

interact with the separate applications.

[0005] Another common characteristic of graphical user interfaces, particularly for those addressing network systems and service management, is that they primarily provide a single view of objects and a single set of capabilities for manipulating the objects they present. Typically there is no means to see objects from different perspectives. For example, if a user is accessing an application to look at the configuration of a computer, they cannot easily see that same computer from the perspective of where it sits in a network and how it communicates with other computers. To do this users typically need to bring up separate applications and take actions to locate the desired object in both. And if the user can see the objects from different perspectives, the different perspectives and related functionality are typically intermixed causing an overload of functionality.

[0006] A somewhat less common aspect of graphical user interfaces is the use of information about the current context to aid the user in performing tasks. For instance, information on where the pointer is located when the mouse is clicked is used to filter pop-up menus. Information on the pointer position or specific location in an application is used to provide context sensitive help. However, there has been no attempt to use context at a higher level to provide users with different information sets or different perspectives on a given information set. User task performance would greatly benefit from an explicit mechanism by which users can designate the perspective they want to take in interacting with a set of applications. Applications could then use this contextual information to determine the type of data and functionality to be presented, the manner in which the data will be presented (e.g., different layouts based on the types of object relationships that are associated with the context), and the default parameters to be used (e.g., systems may be polled every 30 minutes in a system context to make sure they are up and running; however, in an application context they may be polled every 10 minutes to determine if the application is still running).

[0007] There is therefore an unmet need in the art to be able to minimize the number of windows that are required for related applications or application functionality. Additionally, there is an unmet need in the art to allow users to designate a context and have applications share relevant contextual information in such a way that will allow for the presentation of data and functionality based on the perspective associated with that context.

SUMMARY OF THE INVENTION

[0008] It is therefore an object of the present invention to be able to minimize the number of windows that are required for related applications or application functionality in a windows-based computer environment.

[0009] It is a further object of the present invention to

allow users to quickly see objects from different perspectives and to have the visible functionality in a window restricted to that perspective.

[0010] Therefore, according to the present invention, a method and structure for providing contextual information to a user of a windows-based computer environment is provided. After opening a window having a user interface, such as a graphical user interface (GUI), the user can manipulate a context control feature of the user interface to determine the context of the window. The context control feature may be subsequently manipulated to readily define new window contexts as required. This method is capable of being implemented by a computer program of a computer-readable medium. The context of the window refers to a view or views of objects that appear in the window, their relationships, selections made by the user, and capabilities for manipulating the objects. The context control feature may be a context list box, a context tab, or other control feature of the user interface.

[0011] The user interface of the window has at a minimum the context control feature and a content pane. The user manipulates the context control feature to determine a context of the window that is illustrated in the content pane. The user interface may additionally have a scoping pane that provides more detail about the context, thereby allowing the user to specify specific views of the context. Menus of a menu bar and toolbar buttons of a toolbar of the user interface are defined by the context chosen by the user. Thus, when the user changes from a first context to a second context, the scoping pane, menus and toolbar buttons will change as necessary to be in accordance with the second context.

BRIEF DESCRIPTION OF THE DRAWINGS

[0012] The novel features believed characteristic of the invention are set forth in the claims. The invention itself, however, as well as the preferred mode of use, and further objects and advantages thereof, will best be understood by reference to the following detailed description of an illustrative embodiment when read in conjunction with the accompanying drawing(s), wherein:

Figures 1a and 1b illustrate the use of a context list box in an application window that presents views of objects and provides functionality for acting on those objects, according to the present invention;

Figures 2a, 2b, 2c and 2d illustrate the use of a context list box in an application window that allows users to access different types of objects or applications, according to the present invention; and

Figures 3a and 3b illustrate the use of a context list box in an application window that provides lists of

tasks and allows users to access the user interface for accomplishing those tasks, according to the present invention.

DESCRIPTION OF THE INVENTION

[0013] Contextual information is often used to filter information in the user interface of a windows-based computer environment. It is thus useful as a filtering tool to provide information to the user that is consistent with the objects or applications the user is currently using or the perspective from which the user is viewing the objects or applications. The present invention employs the user interface of a single open window, such as an application window, to empower the user to assert direct control over the filtering of information and the perspective from which information is presented to the user in the single window. Through the manipulation of a context control feature of the user interface, the user can choose the desired context for the window. The context is defined such that the objects that appear in the window, their relationships, and capabilities for manipulating the objects are tied to the context. For instance, using the present invention, the user may see the network connectivity of a computer system or the IT services (e.g., email, file sharing) in which the system participates all inside a single window by correctly manipulating the context control feature of the window. The context of the window may be readily changed as desired, again through appropriate manipulation of the context control feature.

[0014] The context control feature, which may be a context list box, a context tab, or other control feature of the user interface, contains a list of the contexts available within the window. User manipulation of context through the context control feature of the window is accompanied by a corresponding change in menu items, toolbar buttons, and views of presented objects according to the context chosen by the user. Relevant information from the initial context, such as currently selected objects, is maintained in the new context of the window and can be used to aid users in quickly accessing task relevant information or functionality. This feature makes the user more efficient in the task at hand.

[0015] Manipulation of the context control feature of the GUI of a single window can enable the user to look at a given set of managed objects in a different perspective or context. Alternatively, changing context can mean the user will have access to a different set of objects. Thus the user can look at a different set of objects from the same perspective as an earlier set of objects.

[0016] First, consider the use of the present invention as an explicit mechanism for filtering information and changing the perspective from which information is presented to the user via the user interface. A change in context via the context control mechanism is typically associated with a change in the objects presented and

the type of capabilities available to the user for manipulating those objects.

[0017] Consider, for example, an application window in which the user can change between the context for the IP Network Management and the context for Service Management using the context control mechanism of the context list box. This use of the context list box is illustrated in Figures 1a and 1b. Referring to Figure 1a, the context list box 12 of the user interface 10 has been set to "IP Internet" and the location cursor is on "Segment 1" in scoping pane 14 so that the context of user interface 10 is IP Internet. The content pane 16 illustrates the IP Internet perspective of objects of Segment 1 of the IP Internet: CItGeorg, nsmdedd-green, et2m41, djm213t, djm23t, and ljm21; note that objects CItGeorg, nsmdedd-green, djm213t, djm23t, and ljm21 are all connected to one another through the network hub et2m41 as indicated by the graphical map configuration shown in content pane 16. In Figure 1b, the same application window with a different context is presented to the user. Context list box 12 has been set to "IT Services" and scoping pane 14 is set to show different types of services so that user interface 20 has a Service Management context. Content pane 16 illustrates the objects involved in the OpenMail service. Note that the selected object "CItGeorg" from user interface 10, has been carried over to show this same object from the new perspective of Service Management. This system is now connected to the mail server "Mail-SV1" that provides email to the system. Notice that available menus 17 and tool bar buttons 18 have been changed to match the new context. By simply manipulating the context chosen in context list box 12, the user explicitly filters information and changes the perspective from which information concerning managed objects is presented to the user.

[0018] Now consider an example in which context is used to view different objects or applications. This use of the context control is illustrated in Figures 2a, 2b, and 2c. In Figure 2a, the context chosen in the application window of user interface 30 is "Users" as shown in context list box 12 and scoping pane 14. In this example, "Users" is the only visible choice in the scoping pane but this item could be expanded by the user to display additional choices. The objects illustrated in content pane 16 include the following user groups: admin, operator, user, MS-SQL, and MS-Exchange. These are the groups of users that are allowed to do backup and recovery of files within the application.

[0019] Typically, using the context control feature, a user will be able to choose from two or more types of context. As illustrated in the user interface 40 of the application window of Figure 2b, by clicking on the context list box 12, the user is presented with a context list 13 showing all of the contexts available in the current window. In user interface 40, users can choose from: Clients, Users, Devices and Media, Backup, Monitor, Restore, Reporting, and Database. In this figure, the

context list 13 is presented in an expanded combination ("combo") box as shown. By selecting one of the contexts in the list, the user moves to that context. If the user selected "Restore" from the context list 13, he or she would be presented with the user interface in Figure 2c.

[0020] In Figure 2c, the context of the application window 50 has been changed to "Restore", as shown in context list box 12 and scoping pane 14. The available objects to restore are shown in content pane 16 and include: Filesystem, OmniBack II database, Rawdisk, and SAP. Figure 2d illustrates another context within the same window. "Devices and Media" is the current context as shown in context list box 12. Selection of this context from the context list (14 in Figure 2b) allows the user to see what backup devices are available and what media have been configured for doing backups. The devices and media are shown in scoping pane 14 of user interface 60. Notice that different menu items 17 are available in the three contexts shown in Figures 2a, 2c, and 2d.

[0021] In addition to being applicable to application windows that display objects, the current invention is equally applicable to application windows that present tasks. Referring now to Figures 3a and 3b, the use of the context list box 12 in a task application window is illustrated. In Figure 3a, the context list box 12 of the user interface 70 has been set to "Service Definition Tasks" and the task "Modify Service" has been chosen in scoping pane 14. The scoping pane in this context shows the hierarchy of tasks that are related to defining the services that an IT group would provide its customers. The selected item "Modify Service" is one of the many different service definition tasks. Content pane 16 illustrates the properties box associated with modifying the service named Order Processing System Access. Figure 3b presents the same task application window "OV Tasks" with a different context. The context list box 12 has been set to OpenView Customization. The change in the context list box results in setting the context for user interface 80 to OpenView Customization. The scoping pane 14 now presents the hierarchy of the tasks associated with customizing OpenView applications. Selecting "New Alarm Wizard" in the scoping pane causes the content pane 16 to present a wizard containing the fields for the data the user must provide to complete the task. Again, by simply manipulating the context chosen in context list box 12, the user explicitly filters information and is presented with that subset of the information that is relevant to the context in the same application window.

[0022] For the figures shown above, a change in the context in which information is presented to users will likely result in a change in the functionality that is available in the menu bar 17 and toolbar buttons 18, the objects, tasks, or tools available in the scoping pane 14 from which the user may choose, and the views that can be presented in content pane 16 of the user interface. It

is envisioned that the information presented in content pane 16 may take any form relevant to the data being presented, for example graphical, tabular, properties, wizard or chart form. The content panes 16 of Figures 1a and 1b, for instance, show information in graphical form; content pane 16 of Figure 1a shows a network map, while Figure 1b shows a service topology. The content panes 16 of Figures 2a, 2b, 2c, and 2d show information in tabular form. The content pane of Figure 3a shows a properties box while the content pane of Figure 3b shows a wizard.

[0023] The present invention therefore provides many advantages to the user. The user can focus on information that is relevant to the current task and this information can in turn be shared by various applications as context filters in order to vary the way that information is presented to the user. This enables the information presentation of the user interface to match the preferences and needs of the user as the user interacts with the computer system. Moreover, the sharing of contextual information allows independently developed applications to appear to the user as highly integrated applications that work closely together within a single user interface window in the computer environment.

[0024] The judicious manipulation of context can also enhance the productivity of the user. Contextual information such as the currently visible view or recent user actions can be used to set defaults that minimize the need for user input. The contextual information can be used to filter irrelevant information or steps so that users can more quickly reach their goals. The passing of contextual information between software within an application and between applications can eliminate the need for the user to duplicate their previous actions.

[0025] While the invention has been particularly shown and described with reference to a preferred embodiment, it will be understood by those skilled in the art that various changes in form and detail may be made therein without departing from the spirit and scope of the invention.

Claims

1. A method for providing contextual information to a user of a windows-based computer environment, said method comprising:

opening a window having a user interface; and

manipulating a context control feature (12) of the user interface to determine a first context of the window.

2. The method of claim 1, wherein manipulating the context control feature comprises selecting a context identifier from a context list (13).

3. A computer-readable medium having a computer

program for providing contextual information to a user of a windows-based computer environment, said computer program comprising:

instructions for opening a window having a user interface; and

instructions for manipulating a context control feature (12) of the user interface to determine a first context of the window.

4. A user interface of a window that provides contextual information to a user of a windows-based computer environment, comprising:

a context control feature (12) of the user interface that has one or more available contexts; and

a content pane (16); wherein manipulating the context control feature (12) of the user interface determines a context of the window from the one or more available contexts that is illustrated in the content pane (16) of the user interface.

5. The user interface of claim 4, wherein the context control feature (12) is a context list box that has the one or more available contexts in a context list (13).

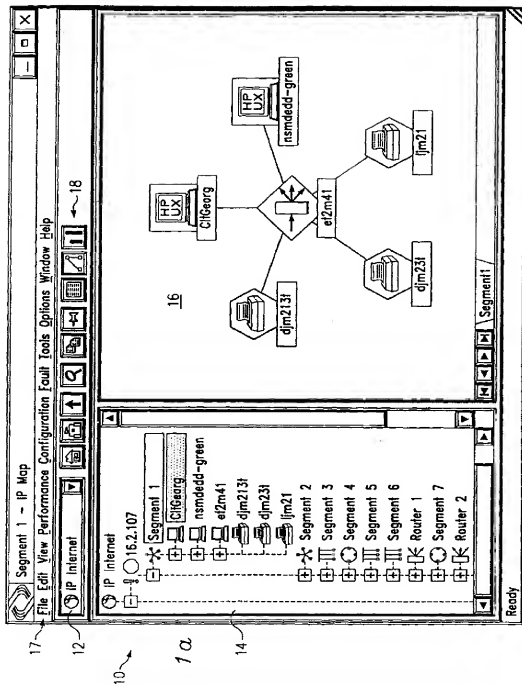
6. The user interface of claim 4, wherein the context control feature (12) is a context tab.

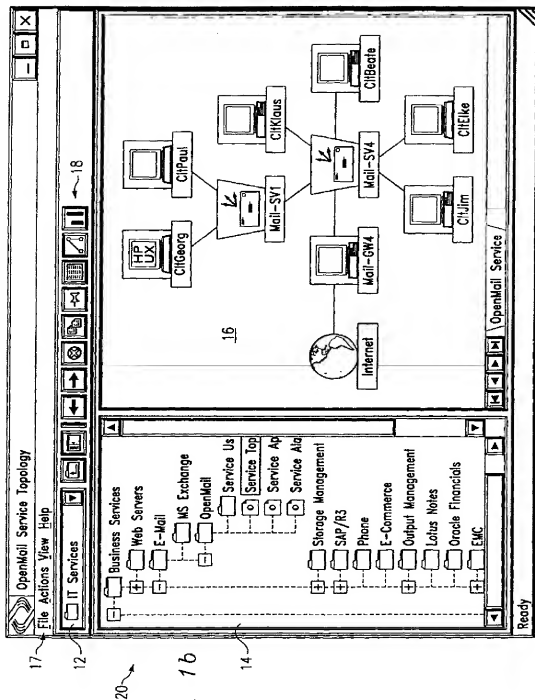
7. The user interface of claim 4, wherein the user interface further comprises a scoping pane (14) that provides a plurality of views of the context.

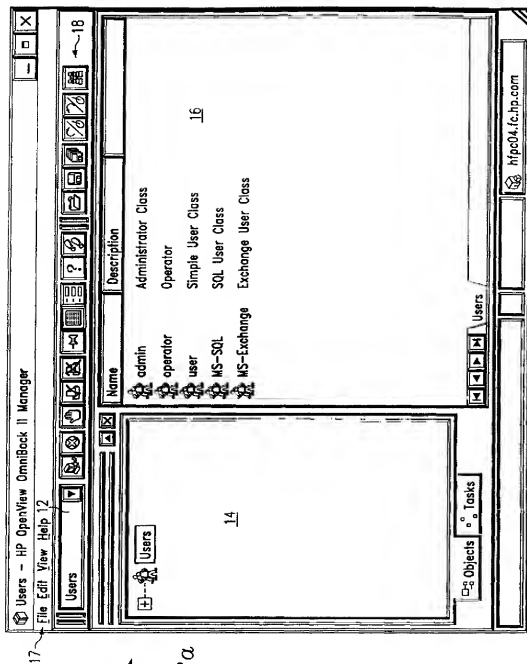
8. The user interface of claim 7, wherein the user interface further comprises:

a menu bar (17); and

one or more static toolbars (18), wherein one or more menus of the menu bar (17), one or more toolbar buttons of the one or more static toolbars (18), and the plurality of views of the scoping pane (14) are defined by the context of the window.







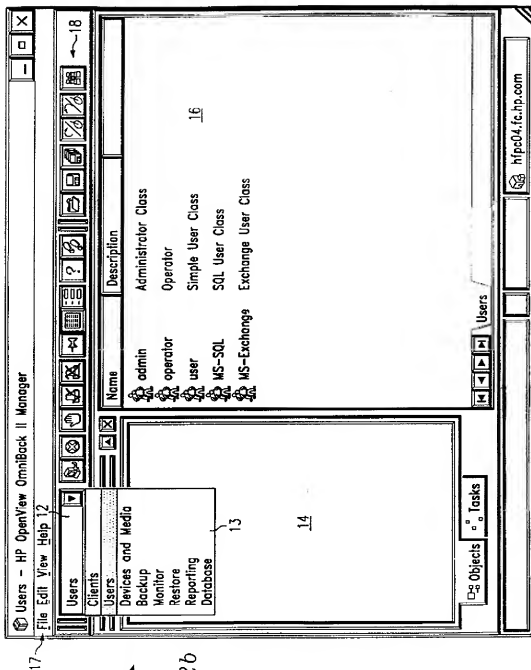
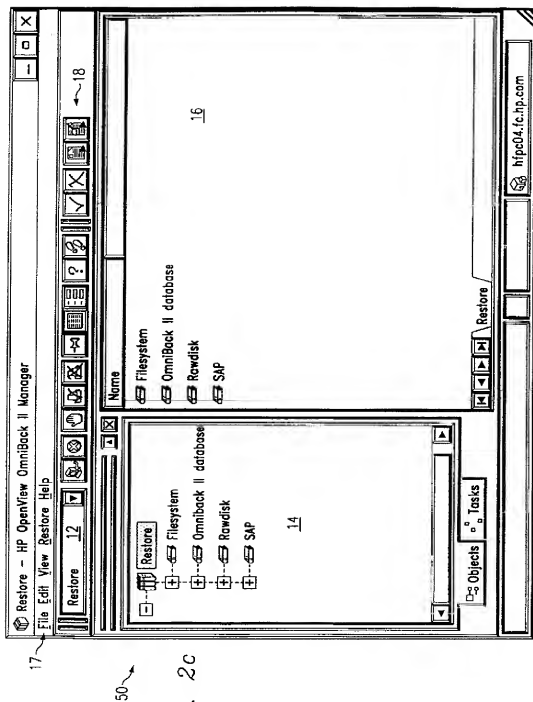


FIG. 2b



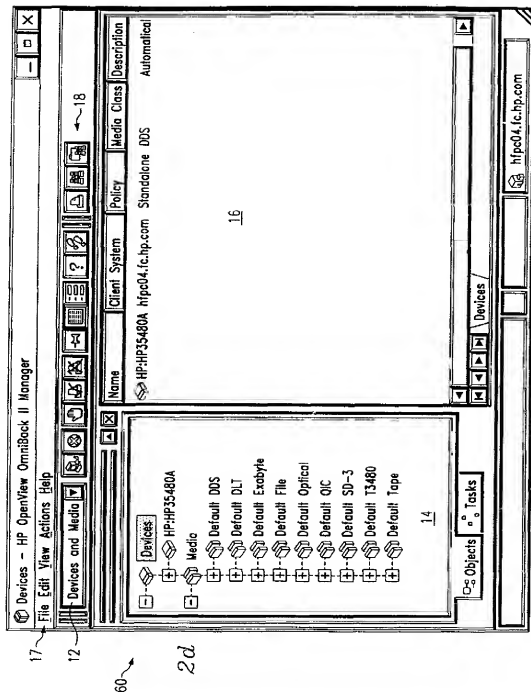


FIG. 2d

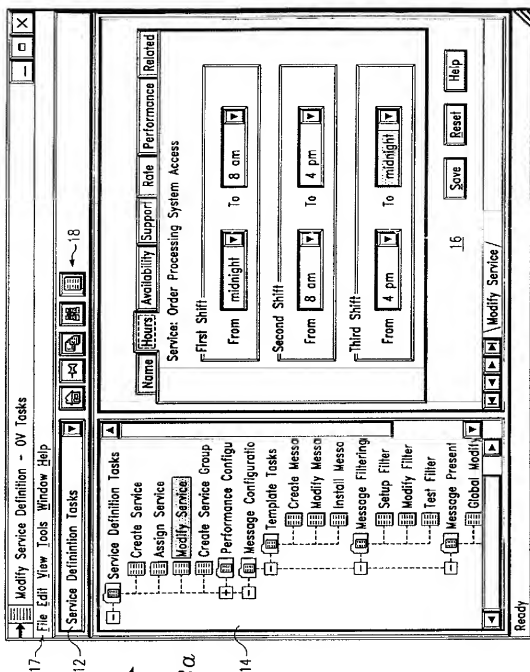
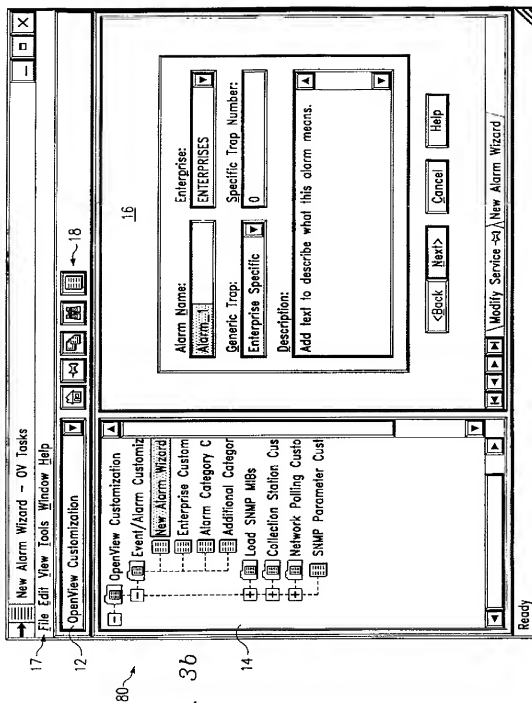


FIG. 3a



European Patent
Office

EUROPEAN SEARCH REPORT

Application Number
EP 96 12 3600

DOCUMENTS CONSIDERED TO BE RELEVANT			
Category	Citation of document with indication, where appropriate, of relevant passages	Relevant to claim	CLASSIFICATION OF THE APPLICATION (Int.Cl.6)
X	EP 0 489 576 A (SONY CORP AMERICA) 10 June 1992 (1992-06-10) * page 5, line 13 - line 27; claim 1; figure 4 *	1, 3, 4	G06F3/033
X	EP 0 438 017 A (IBM) 24 July 1991 (1991-07-24) * column 4, line 36 - line 47 * * column 5, line 1 - line 29; figures 2-6 *	1, 3, 4	
A	US 5 572 648 A (BIBAYAN FARZAD) 5 November 1996 (1996-11-05) * abstract; claims 1, 2; figures *	1, 3, 4	
The present search report has been drawn up for all claims			TECHNICAL FIELDS SEARCHED (Int.Cl.6) G06F
Place of search BERLIN		Date of completion of the search 6 September 1999	Examiner Durand, J
CATEGORY OF CITED DOCUMENTS X : particularly relevant if taken alone Y : particularly relevant if combined with another document of the same category A : technological background O : non-written disclosure P : intermediate document T : theory or principle underlying the invention E : earlier patent document, but published on, or after the filing date D : document cited in the application L : document cited for other reasons & : member of the same patent family, corresponding document			

EPO FORM 1662 (08/99) (P4201)

**ANNEX TO THE EUROPEAN SEARCH REPORT
ON EUROPEAN PATENT APPLICATION NO.**

EP 98 12 3600

This annex lists the patent family members relating to the patent documents cited in the above-mentioned European search report. The members are as contained in the European Patent Office EDP file on
The European Patent Office is in no way liable for these particulars which are merely given for the purpose of information.

06-09-1999

Patent document cited in search report		Publication date	Patent family member(s)	Publication date
EP 0489576 A		10-06-1992	US 5148154 A	15-09-1992
			DE 69131065 D	06-05-1999
			EP 0786716 A	30-07-1997
			EP 0786717 A	30-07-1997
			JP 4307675 A	29-10-1992
			US 5307456 A	26-04-1994
EP 0438017 A		24-07-1991	US 5347628 A	13-09-1994
			DE 69023755 D	04-01-1996
			DE 69023755 T	20-06-1996
			JP 3222033 A	01-10-1991
			JP 7027341 B	29-03-1995
US 5572648 A		05-11-1996	JP 6348450 A	22-12-1994

EP 0 961 197 A1

For more details about this annex : see Official Journal of the European Patent Office, No. 12/92